NORTH EAST FLORIDA T/S USERS GROUP

THE SINC TIMES

VOL 1 ISSUE 1 $1.00 JAN 1985

EDITORIAL BY STEVE BERG

FIRST AND FOREMOST, WELCOME TO OUR FIRST ISSUE OF THE SINC TIMES. THIS IS AN EXCITING TIME FOR US. WE HAVE SOME REAL TALENTED PEOPLE IN OUR SUPPORT GROUP AND AS YOU WILL SEE FROM THIS NEWS LETTER THEY ARE WILLING TO SHARE THIER IDEAS AND THOUGHTS WITH US. IN THE FUTURE WE HOPE TO HAVE EVEN MORE PEOPLE BECOME CONTRIBUTORS, NOT ONLY FROM OUR GROUP BUT FROM OUR READERS ALSO.

IN THIS NEWSLETTER WE WILL TRY TO SUPPORT ALL SINCLAIR AND TIMEX/SINCLAIR COMPUTERS. SINCE TIMEX PULLED THE PLUG ON US, SEEMINGLY AT THEIR PEAK, WE MUST NOW SUPPORT OUR SELVES BY SUPPORTING ONE ANOTHER. THIS PUBLICATION IS FOR THAT PURPOSE. READ IT, ENJOY IT, THEN PASS IT ON. AGAIN WELCOME TO THE SINC TIMES.

STEVE BERG
(EDITOR)

HAVE COMMENTS/SUGGESTIONS?
MAIL TO:

THE SINC TIMES
6634 ORIOLE AVE
JACKSONVILLE, FL 32216

ATTN: STEVE BERG

ABOUT THE SINC TIMES

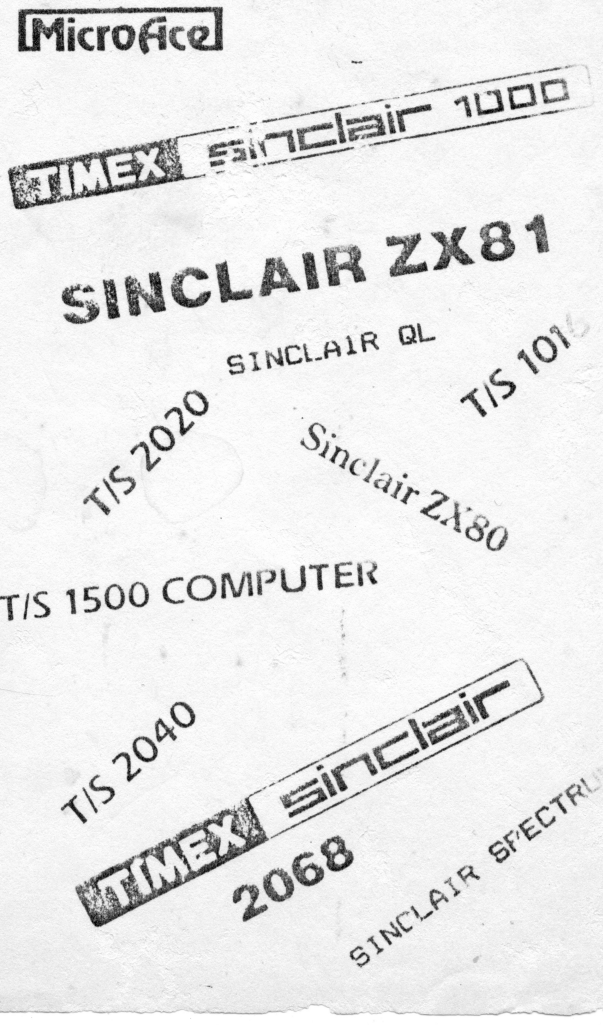
EDITOR----------STEVE BERG
CO-EDITOR-------JOHN KUHN

THIS MONTHS CONTRIBUTORS-

JOHN KUHN
STEVE BERG
BILL FISHER
DOUG CARTER
PERRY WILSON

INDEX

		PAGE
Editorial	.....	1
TRIG 2	.....	2
BILL's BYTES	.....	2
Editor's Notes	.....	2
Micro Ace / ZX-80 Program	.....	2
TS 1000 MACHINE CODE PROGRAMMING	.....	3,
COMPUTERS FOR FINANCES	.....	5
TS NEWS TO USE	.....	6-
BUD's MUD	.....	1
Book Reviews	.....	1
NEF TS USERS Stats.	.....	1



WE ARE LOOKING FOR SOME GOOD ZX-80 PROGRAMS TO USE IN THIS NEWSLETTER. ANY GOOD ORIGINAL PROGRAMS YOU HAVE TO SHARE PLEASE CONTACT THE SINC TIMES IN CARE OF THE EDITOR

THANKS

MATH FOR THE 2068 - TRIG 2
BY PERRY WILSON

TRIG 2

```
  20 PRINT AT 1,13;"TRIG 2";
  30 PRINT
  40 PRINT "LET  G  REPRESENT DE
GREES"
  60 PRINT "ENTER G"
  70 INPUT G
  90 PRINT AT 4,19;G
 100 LET X=INT (G/360)
 110 LET Q=G-(X*360)
 120 PRINT AT 6,5; X ;" CIRCLE(S
) + ";Q;" DEG."
 130 IF Q>0 AND Q<91 THEN PRINT
AT 7,5;Q;" DEG. IN QUADRANT 1"
 140 IF Q>90 AND Q<181 THEN PRIN
T AT 7,5;180-Q;" DEG. IN QUADRAN
T 2"
 150 IF Q>180 AND Q<271 THEN PRI
NT AT 7,5;Q-180;" DEG. IN QUADRA
NT 3"
 160 IF Q>270 AND Q<361 THEN PRI
NT AT 7,5;360-Q;" DEG. IN QUADRA
NT 4"
 170 PRINT AT 9,3;"SIN";TAB 23;"
COS";AT 12,3;"TAN";TAB 23;"COT"
 180 LET C=PI/180
 190 PRINT AT 10,0;SIN (G*C);TAB
 19;COS (G*C);AT 13,0;TAN (G*C);
TAB 19;1/TAN (G*C)
 200 PRINT AT 15,3;"SEC";TAB 23;
"CSC"
 210 PRINT AT 16,0;1/COS (G*C);T
AB 19;1/SIN (G*C)
 220 PRINT AT 18,12;"RADIANS"
 230 PRINT AT 19,11;(G*C)
 240 PRINT AT 21,0;"TO PROCEED,
PRESS RUN, ENTER -- TO EXIT ; PR
ESS X, ENTER."
```


BILL'S/BYTES:
BY BILL FISCHER

WITH THIS BEING OUR FIRST NEWSLETTER FROM THE N.E.F. T/S USERS, I WOULD LIKE TO SEND A BIG WELCOME TO ALL OF OUR USER GROUP MEMBERS. THEY ARE AS FOLLOWS;

JERRY BARKER
STEVE BERG
WARD BIRKEY
ED DAVIS
BILL FISCHER
JOHN KUHN
KEITH LANTZ
MATT MATHIESON
CHARLES MURPHY
DAVE PEYTON
TOM PHILLIPS
JOHN SHEETS
PAT WILLIAMS
PERRY WILSON

IN THE MONTHS TO COME WE ARE LOOKING FORWARD TO SEEING ALOT OF NAMES OF NEW N.E.F. T/S USER GROUP MEMBERS.

EDITOR'S NOTES By Steve Berg

UPCOMING IN THE SINC TIMES
reviews of> VU 3D
VU FILE
VU CALC
MUSICOLA
PENETRATOR
TASWORD TWO
SPELLING 1
FLIGHT SIMULATOR
HOT Z
and hardware reviews on;
THE BYTE BACK MODEM 2068 versi
THE TIMEX MODEM with SMARTERM
SOFTWARE.
GORILLA BANANA PRINTER
EPSON MX-80 PRINTER
AERCO DISC INTERFACE (TS 1000)

and LOTS LOTS MORE.

HAVEYOUHUGGEDYOURCOMPUTERTODAY

TS 1000 MACHINE CODE PROGRAMMING

by Doug Carter

I have been asked to write a little article about my experiences with machine code. I have very limited experience with programming and next to none with machine code so forgive me if I am too basic.

My first discovery was, like most, a product of necessity. I had a machine code program in PROM that wouldn't run as advertised. My printer wouldn't work, my disassembler wouldn't run so I had to do something drastic. Mat Mathieson suggested that I "block move" the entire program with machine code into user RAM. That would allow me to SAVE the program to tape and move it to another machine. He also lent me his copy of Zacs' "How to program the Z80". Armed with this manual and a lot of verbal support from Mat, I proceeded to learn "block move".

Before we move on to machine code programming, let's look at Clives' storage method. In chapter 26 of the User Manual, we find that data is stored in a predetermined format. Page 128 shows us that there are two bytes for the line number, two for the length, then the text. On that same page, they show us that the program area starts at 16509. We put these facts together and add a little math, we can predict where anything is in RAM.

The "block move" is a machine code method of moving data from one point in memory to another. The only parameters are that you know where in memory the data is located, how many bytes are to be moved, and where you want the data to end up.

In my example, I have a program in PROM, the program resides at location 8192, is 2048 bytes long, and I want it in a "1 REM" condition in RAM. The first step is to create a machine code program that will perform the required move.

The Z80 has several moves available, the one I chose is the "LDIR". This is a repeating block move with increment. The requirements are that the memory location in the HL register is the "move from" location, the memory location in the DE register is the "move to" location, and the number of bytes to be moved is in the BC register.

Now for a program, we have to get the data into the several registers and then perform the move. Zaks explains that To place data into the HL register, use LD HL (the mnemonic being represented by the decimal number 033); the DE register, use LD DE (017d); the BC register, LD BC (001d). The addresses and numbers that you are loading are entered as two bytes, the first being the least significant. To explain further, divide the number by 256, the whole number dividend is byte two, the remainder is byte one.

I want to clarify the addressing a little. I'll use a road map to help;

```
16509  Two bytes of line number, most significant first
16511  Two bytes of line length, least significant first
16513  The token REM
16514  2048 bytes of space to move program into
18562  The token ENTER
18563  Two bytes of line number, same as before
18565  Two bytes of line length, same as before
18567  The token REM
18568  12 bytes of move program
```

So, to review, we must load the three registers and perform a LDIR, a two byte instruction (237,176d). The last step is a return to basic (201d). If we put this all together, we end up with the following;

LINE #	MNEMONIC	DECIMAL #	PROGRAM DATA
1	LD HL	8192	033, 000, 032
2	LD DE	16514	017, 130, 064
3	LD BC	2048	001, 000, 008
4	LDIR		237, 176
5	RET		201

Now that we have our program, we're ready to load it into the computer. I use REM Builder (see Sync Volume 3, Issue #6, Page# 72) to build a 1 REM of 2048 bytes to receive the moved program. Next, enter a 2 REM with 12 spaces.

To load your data, build a "FOR/NEXT" routine to poke your input in the twelve memory locations.

```
10 FOR X = 18568 TO 18579
11 INPUT A
12 POKE X, A
13 PRINT X, A
14 NEXT X
```

Start it with GOTO 10 and enter the data as required. Now add program line 3; 3 RAND USR 18568 Rather than erase your REM loader routine, just add a line 4; 4 STOP.

The program is all ready to use. I hope that I have made it understandable for everyone. With a little thought, you can change a few points and move things above RAMTOP quite simply. A basic program to move data above RAMTOP is, at best, slow and cumbersome. If anyone learns one new point, I've done what I set out to do.

COMPUTERS FOR FINANCES
By Steve Berg
Here is a short program that figures loan payments, interest, etc. for the 2068. It can easily be converted to use on the T.S. 1000 or zx/81.
Note that interest is broken down and figured on a monthly basis rather than annually. Interest must be entered as a decimal. i.e...12 for 12%.
This is a very basic program and has lots of room for improvement. In the forth coming issues of the SINC TIMES we will explore things like cleaning up the INPUT, crash proofing, tidying up the display and general streamlining of the program. If you have comments or questions Please contact me at editor's address.

```
  5 REM "LOAN"
 10 PRINT "AMOUNT OF LOAN"
 20 INPUT AMT
 25 PRINT AMT
 30 PRINT "ANNUAL INTEREST RATE ?"
 40 INPUT RAT
 45 PRINT RAT
 50 PRINT "LOAN DURATION (months)?"
 60 INPUT MO
 65 PRINT MO
 70 LET INT=(RAT/100)/12
 80 LET DIVISOR=1+INT
 90 LET DIVISOR=DIVISOR^MO
100 LET DIVISOR=1-(1/DIVISOR)
110 LET P=AMT*(INT/DIVISOR)
120 PRINT "PAYMENT=";P
130 PRINT
140 PRINT "NO. INT PAID REDUCTION BALANCE"
145 FOR I=1TO MO
150 LET INTPAID=AMT*INT
160 LET RED=P-INTPAID
170 LET AMT=AMT-RED
190 PRINT I;TAB 4;INT (INTPAID*100+.5)/100;TAB 13;INT (RED*100+.5)/100;TAB 28
NT (AMT*100+.5)/100
200 NEXT I
```

T S N E W S T O U S E

BY John Kuhn

Welcome to VOL.#1 ISSUE #1 of the SINC TIMES. In my little NTU (NEWS TO USE) corner I will attempt to pass along information from various sources on programming tips, new Hardware/Software, and other developments that directly affect us ZX-TS survivors.

Over the years of using the ZX-TS machines, I have accumulated some helpful programming tips that I have compiled into my First NTU column. Most likely you will have run across many of these tips before from fellow programmers or NEWSLETTERS / MAGAZINES, but hopefully many of you will gleen one or two useful bits of new information to help sharpen your programming skills.

An "*" beside the TIP # will indicate that this tip pertains only to the TS 2068 computer.

---

1) *TRUNCATE & ROUND OFF OUTPUT*

Since the TS computers do not have a command to preset the number of decimal places that will be output, the following little routine will do it for you. Example: If you ask your machine to print the value of 22/7 you get 3.1428571 or a number with 7 decimal places. The 7 decimal places may look quite impressive, but try to print it at column #28 and you will overwrite the next 5 spaces in the following line. SOLUTION: Try this. Before you print X to the screen, add a line that says :

LET X = INT ((X+.5/ 1Ey)* 1Ey)/ 1Ey where y= the # of decimal places that you would like X to be truncated and rounded off too. In the above example set y=2 so the answer will print out as 3.14.

---

* 2) To input a value on lines #22 & #23 so as not to overwrite text on lines #1 thru #22 use this format :

[INPUT "YOUR STATEMENT "; X] where X can be either a numerical or string variable.

---

* 3) To prevent the stop command from functioning on a string input in a program that does not use the ""ON ERR"" command, try this:

INPUT LINE X$ or INPUT "YOUR STATEMENT "; LINE X$. (LINE command is on the EXTENDED mode key 3). One feature you will notice when using this technique is that the quotation marks will be missing around the input cursor even though you are entering a string variable. This prevents you from deleting the left quotation mark and inputting STOP. Unfortunately an apparent BUG in the TS 2068 ROM will STOP the program if CAP SHIFT #6 is pressed.

---

4) *STOPPING SELF RUNNING PROGRAMS ON ZX-81/ TS 1000's.*

To stop any ZX-81/ TS 1000 program from AUTORUNNING so you can make backup copies or examine the program, try this:

LOAD CHR$ USR 832 "" instead of just using LOAD "". After the program goes thru the normal load pattern it will return to a Grey search screen. PRESS BREAK at this time and then press LIST + ENTER. Save as usual or GOTO the line # that has the SAVE command to make a self running duplicate. The advantage to this procedure is that it does not take any memory or require you to load in a stopper program.

---

5) *SAVING RAM WHILE PROGRAMMING*

This subject alone could be the basis for a complete series of NTU articles so I will only cover three short tips.

A) Remove unneeded REM statements *[make sure not to remove those that have machine language in them].*

B) Use KEYBOARD words in your print statements.

Eg.: To enter the following statement,

10 PRINT "INPUT VALUE" Try this.

The word INPUT is the keyword on the (I) key. First enter line # 10 then press the keyword INPUT. Now use the shifted #5 key to backspace to the left of the word INPUT and press the P key to now have 10 PRINT INPUT. Now add quotes to the left of the word INPUT and use the shift #8 key to move to the right of the word INPUT. Now type in the word VALUE" & press ENTER. Instead of the word INPUT taking 5 bytes of memory it will now only take 1 byte.

C) USE of the VAL COMMAND:

Eg. Instead of using a statement like 100 GOSUB 6000 try using: **100 GOSUB VAL "6000".**

The use of VAL will yield a net savings of 3 BYTES of memory saved per usage. One trade off in using the VAL command is that program execution speed is slower.

The above techniques are especially useful on the ZX-TS 16K or less machines, but why would you worry about saving memory on the TS 2068? ANSWER: DATA BASE programs will have more memory for data, SPREADSHEETS will have more room for number crunching and WORD PROCESSORS will have a bigger text file area.

---

* 6) *PROGRAM FILE*

Unlike the the ZX-81 or TS 1000 whose program file always begins at 16509 the TS 2068 does not have a set starting address. It's starting address may be found by: **PRINT PEEK 23635 + 256 * PEEK 23636.**

*** 7) If you plan to use UDG's (USER DEFINED GRAPHICS)** in a long BASIC program that will incorporate a Video Mode Change, first lower RAMTOP to 63255 by using **CLEAR 63255.** A BUG in the TS 2068 ROM will allow a long BASIC program to overwrite your UDG's if RAMTOP is not first lowered.

*** 8) If you are using cartridge software** that can be stopped by using the break key, you may enter your own BASIC lines into RAM by first **POKING 23750,0.** If you would like to return to the cartridge ROMware **POKE 23750,128**

---

9) Since the use of the INT command always rounds the subsequent # down, use this instead to give a better rounded off figure. INT (X + .5)

---

* 10) You may switch from CAPS LOCK to LOWER CASE characters (and back again) within a program by using POKE 23658,8 (CAPS LOCK) or POKE 23658,0 (LOWER CASE).

---

* 11) When using the PRINT command the use of ' will give a line feed on the PRINT statement. Eg. PRINT ''' "Hello" will print Hello on line #4.

---

* 12) The TS 2068 is one of the few MICRO computers that can normally only input information on the bottom two lines of the screen, however the following little trick will allow you to input information anywhere on the screen. NOTE: This routine will scroll other information off the screen if it is not used first.

```
EG.: 5 PRINT AT 10,10;" THIS IS PRINTED AT 10,10"
     10 INPUT AT 22,0; AT 0,0; "INPUT #1 ";A$; AT 1,0;
     "INPUT #2 ";B$; AT 2,0; "INPUT #3 ";C$ .... etc.
```

Line #5 was only added to show you what will happen if other information is on the screen when you use the INPUT AT command.

---

13) As you probably know, using the NEW command will only clear out the memory below RAMTOP. Occasionally though I find myself needing to completely RESET the computer memory for one reason or another. Normally in the past I have gotten in the habit of switching the power off then back on to accomplish this. Not only does this tend to wear out the on off switch, but more importantlly each time the power is recycled a power surge hits the computer. SOLUTION: Type PRINT USR 0 will reset computer the same as if it were cold started.

---

* 14) To obtain an audible feedback from the TS 2068 keyboard when typing or programming POKE 23609,X where X can be any number from 1 to 255. Poking different numbers will yield different sounds. (I personally like to POKE 23609,12)

---

* 15) Use USR 15000 in a VARIBLE INPUT statement (will not work with a string variable INPUT statement) to STOP a program that has an ON ERR command that prevents the use of the BREAK key from functioning. After program stops then IMMEDIATELY type ON ERR RESET to deactivate the ON ERR command and prevent keyboard lockup. A problem that I have run into using this method is that if a particular program has its own character set above RAMTOP for the computer to use you will oftentimes get a garbage listing of the program. If this happens POKE 23606,0 & POKE 23607,60 to redirect the computer to the ROM area character set to obtain a program listing. Unfortunately the alternate character set will oftentimes be destroyed by using this trick.

---

* 16) To change the amount of time needed to hold down a key until it repeats **POKE 23561,X where X may = 1 to 35.** Initially the value will be 35 on power up, but each lower value will speed up by 1/60th of a second the time until the key repeats.

---

* 17) To change the amount of time between consecutive repeats (once key starts repeating) **POKE 23562,X where X = 1 to 5.** This also speeds up function by 1/60th of a second for each lower number entered.

*** 18) Address 23692** contains the value that the computer uses as **SCROLL CONTROL.** Poking any number from 1 to 255 into this address will cause the computer to scroll that many lines + 21 on the screen.

---

* 19) If you have created your own set of **UDG's (User Defined Graphics)** and would like to save them to tape so you can reload them at will without having to either remake them or load in a program that will remake them then try this: **FIRST type SAVE "name" CODE USR "A",21*8.** If you have not created a full 21 characters then substitute the number of characters you wish to save for the 21 in the above statement. Now to reload your custom character set:**Type LOAD "" CODE USR "A",21*8** and you will easily be able to load your UDG's into your own programs.

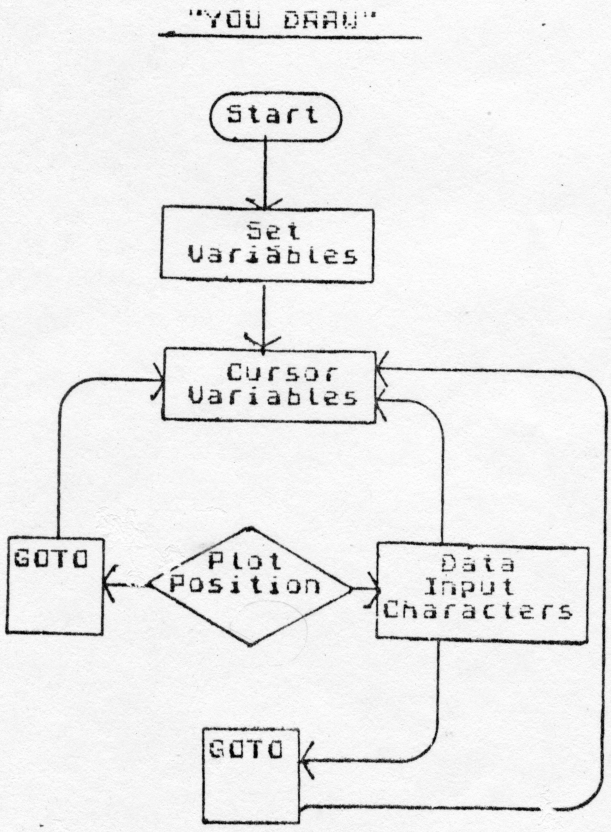
I wish I had more room in this first installment of **TS NEWS TO USE** but newsletter space dictates that I must at minimum make this a multipart series. Part II will be in VOL. #1 ISSUE #2 of THE SINC TIMES. If you have any NEWS TO USE tips please drop me a line with them & I'll see about including them in an upcoming NEWS TO USE column. Until the next time keep on ZX-TS computing.

SEND TO: JOHN KUHN
N.E.F.T.S.U.G *TS NEWS TO USE*
1707 KING ST.
JACKSONVILLE, FL. 32204

BUD'S MUD
by Bill Fischer

This is the first of a series of articles that will be filled with some crazy ideas and programs which will allow you to make some good use of your Timex /Sinclair computers and maybe bring you a little enjoyment a-long the way. I have been work-ing on a program to put this first article, however I have been having some problems with a bug that has gone into my pro-gram and has caused everything to go haywire. So I guess that you are just going to have wait until next month to get my first program.

Well maybe some of you out there can give me a he lping hand in finding the little bug that has been giving me so much trouble. The following is the flowchart that I am using for my program:



If there are any of you programs out there that can give me a hand on developing this program, please let me know. For those of you that are not into programing I will try to have my program done by next month's article. This is the friendly bug chaser saying "See you next month"....

(ED. NOTE...AS SOON AS WE CLEAR AWAY THE MUD MAYBE WE CAN FIGURE OUT THIS ARTICLE AND WHERE ALL THE BUGS CAME FROM!)

TS 1000 BOOK REPORT

Timex Sinclair BASIC Primer with Graphics by Waite and Chapnick

This is an excellent beginners book. It gives you a good foundation to build your BASIC programing skills on. The usual how to set chapters are at the beginning and look like every other TS 1000 book but before long you find the later chapters are really excellent reading and learning material. There are extra chapters devoted just to TS 1000 graphics that can be of use to even the experienced programmer. This is a good book and is recomended for your library . Its available from SAMS Publishing.

A brief note about the NEF TS USERS .
Our officers are;

Steve Berg	President
Bill Fisher	Treasurer
Pat Williams	Librarian
JOHN KUHN	SECRETARY

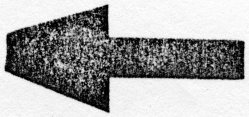
Dues are $10.00 Per year.

The meetings are on the second and forth Saturdays of each month. We are currently meeting at the city owned Buckman Plant near Haines and 8th Streets.We thank the city for the use of the lunch room at this facility. This area was chosen for its central location in the city. For instructions to get to the meeting Place call 737-1538.

TS 2068 Beginner/Intermediate Guide Book Report
If you have been looking for the advanced programmers guide talked about by Timex you just found it. This book is the 1st half of that guide. If you REALLY want to know that 2068 sitting on your desk this book is the place to start. It gives you a good foundation to build your overall computer knowledge on as well as the specifics you need for your 2068. This book and it's mate,the intermediate/ advanced guide are a MUST for the 2068 user. Available from SAMS Publishing.

Opinions expressed in this news letter are those of the author of the particular article and are not necessarily those of the editors or the NEF TS USERS.

In THE SINC TIMES it is the sole respondsibility of the contributing authors to insure the no copyright infringements occur in their respective contributions. NEITHER THE SINC TIMES NOR NEF TS USERS are re-spondsible for any copyright infringements. Original material printed in this newsletter may be reprinted without permission by other TS USER groups and educational institutions only. An aknowledgement of the THE SINC TIMES and the author must be included. ALL OTHERS must first obtain permission before using any material from THE SINC TIMES.



For those of you that haven't heard it looks like the TS 2068 will be back on the market. It will be under a new name . This is based on what we have read in some other newsletters that we believe to be quite reliable . How would you like to see "THE LITTLE COMPUTER STORE" in JAX,FL.